Using Python for Accelerator Data Acquisition

Beau Harrison

25th June 2019

Download Libraries Locally

Three files are needed to make DPM requests.

- Retrieve files from Redmine
  - https://cdcvs.fnal.gov/redmine/projects/acnetd/repository/python
    - acnet.py
  - https://cdcvs.fnal.gov/redmine/projects/acsys-dpm/repository/py-client
    - DPM.py
    - dpm_protocol.py
  - A branch for each of the Python versions
  - Use git to clone the repositories or download them by hand

Make a Basic Request

```python
#!/usr/bin/env python2
import DPM

dpm = DPM.Blocking()
dpm.addEntry(0, 'M:OUTTMP@e,8f')

for ii in dpm.process():
    print ii
    print "Doing work..."
```

```python
#!/usr/bin/env python3
import DPM

dpm = DPM.Blocking()
dpm.add_entry(0, 'M:OUTTMP@e,8f')

for ii in dpm.process():
    print(ii)
    print("Doing work...")
```

Details of a Request

- Choose a data request strategy
  - Takes two optional arguments
    - ACNET connection for multiple services
    - task@node string for testing
  - Blocking – `DPM.Blocking()`
    - Synchronous – data driven
    - Get data -> do something with data
    - Similar to standard for loop
  - Polling – `DPM.Polling()`
    - Asynchronous – logic driven
    - Initialize request -> process all data since last processing
    - Necessary if your coding is doing other things
- Add requests to your list – `dpm.add_entry(9, "M:OUTTMP")`
  - First argument is the reference id
    - Choose a unique int to track your requests
  - Second argument is a DRF string
    - www-bd.fnal.gov/controls/public/drf2/
    - Not in DRF2
      - Data source
        - `FTP`
        - `SRFILE`
        - `LOGGER`
        - `LOGGERPOINT`
        - `LOGGERDURATION`
        - `REDIR`

Details of a Request

- Start DPM request – `dpm.process()`
  - Begins data requisition
  - Accepts an optional data source string
    - "`FTP`" – Plan to auto detect this
      - Provides arrays of data and time us
    - "`LOGGER:T1:T2:NODE`"
      - Times are in UTC ms
    - "`LOGGERPOINT:T1:WINDOW:NODE`" - WIP
      - Time is in UTC ms
      - Window is in ms
    - "`LOGGERDURATION:DELTA:NODE`"
      - Delta is in ms before now
    - "`SRFILE:FILENUMBER`"
      - Provides one data point at the SR time
    - "`REDIR:NODE->NODE`"
      - Allows requests to be redirected to a specific node
- Handle data
  - `dpm.process()` will always have the next reply until you exit the loop
  - `dpm.add_entry()` can be changed on the fly
    - Will not take effect until a new request is started with `dpm.process()`
  - Data is a dictionary
    - Keys of: `tag`, `stamp`, and `data`
    - `tag` matches your selected int

A (Somewhat) Practical Example

```python
#!/usr/bin/env python3
import DPM
import time
from datetime import datetime
from datetime import timedelta

dpm = DPM.Blocking(None, "VIRT01")
dpm.add_entry(0, "M:OUTTMP<-LOGGERDURATION:" + str(1000 * 60
))
data_log = []

for data in dpm.process():
    if data.data != []:
        for datum in data.data:
            data_log.append(datum)
    else:
        break

print(data_log)
```

```python
dpm.add_entry(0, "G:SCTIME")

for datum in dpm.process("SRFILE:990"):
    print(datum)
    break

dpm.add_entry(1, "M:OUTTMP@E,00")
dpm.add_entry(2, "G:SCTIME")
time.sleep(1)

end_time = datetime.utcnow() + timedelta(seconds=61)
for datum in dpm.process():
    print(datum)
    if datum.stamp > end_time:
        break
```

Asyncio Proposal

```
#!/usr/bin/env python3
import DPM
import asyncio

dpm = DPM.AsyncIO()
dpm.add_entry(0, 'M:OUTTMP@e,8f')


async def main():
    async for datum in dpm.process():
        print(datum)
        print("Doing work...")

asyncio.run(main())
```

Other Services Not Yet in Python

- Sync
  - Clock event notifications
- ACL
  - Allows ACL commands to be passed to Centra
- Lookup
  - Cached device database, super fast
- Settings - soon
  - Robust authentication with Yubikeys in the works
  - Limited to CLX nodes at the moment

Subscribe to the Mailing List

- Send an e-mail message to listserv@fnal.gov
- Leave the subject line blank
- Type "SUBSCRIBE CONTROLS-PYTHON-NEWS FIRSTNAME LASTNAME" (without the quotation marks) in the body of your message.

- Any questions or suggestions can be asked on the list or directed to
  - beau@fnal.gov
  - kingc@fnal.gov
  - neswold@fnal.gov

- Resources
  - https://cdcvs.fnal.gov/redmine/projects/acsys-dpm/wiki/Python_DPM_API
  - https://cdcvs.fnal.gov/redmine/projects/acsys-dpm/wiki/Data_Sources